



GİRESUN
ÜNİVERSİTESİ



FEN BİLİMLERİ ENSTİTÜSÜ

**ARIMA MODELLERİNDE
PARÇACIK SÜRÜ OPTİMİZASYONU
İLE PARAMETRE TAHMİNİ**

**MATEMATİK
ANA BİLİM DALI
Yüksek Lisans Tezi
Ali KARADAĞOĞLU
162110012
Haziran 2019**

GİRESUN

**T.C.
GİRESUN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**ARIMA MODELLERİNDE
PARÇACIK SÜRÜ OPTİMİZASYONU
İLE PARAMETRE TAHMİNİ**

YÜKSEK LİSANS TEZİ

Ali KARADAĞOĞLU

Enstitü Ana Bilim Dalı : Matematik

Tez Danışmanı : Prof. Dr. Erol EĞRİOĞLU

Haziran 2019

T.C.
GİRESUN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ARIMA MODELLERİNDE PARÇAÇIK SÜRÜ
OPTİMİZASYONU
İLE PARAMETRE TAHMİNİ

YÜKSEK LİSANS TEZİ

Ali KARADAĞOĞLU

Enstitü Anabilim Dalı :Matematik

Bu tez 27/06/2019 tarihinde aşağıdaki jüri tarafından oybirliği ile kabul edilmiştir.


Doç.Dr.
Cem KOÇAK

Jüri Başkanı


Doç.Dr
Eren BAŞ

Üye


Prof.Dr.
Erol EĞRİOĞLU

Üye

Doç.Dr.
Bahadır KOZ
Enstitü Müdürü

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

Ali KARADAĞOĞLU

23/07/2019



TEŞEKKÜR

Tez çalışmamın her safhasında yol gösteren ve yardımlarını esirgemeyerek karşılaştığım her zorluğun üstesinden gelmemde yardımcı olan çok değerli tez danışman hocam sayın Prof. Dr. Erol EĞRİOĞLU 'na en içten saygı ve teşekkürlerimi sunarım.

Çalışmalarım sırasında bilgi ve tecrübelerinden yararlandığım başta sayın Doç. Dr. Eren BAŞ olmak üzere Giresun Üniversitesi İstatistik Bölümü'nün çok değerli hocalarıma sonsuz saygı ve sevgilerimi sunarım.

Çalışmalarım boyunca yardımını hiç esirgemeyen değerli arkadaşım Oğuzhan YILMAZ 'a da teşekkürlerimi bir borç bilirim.

Ayrıca bu süreçte yanımda olan, desteklerini esirgemeyen aileme ve arkadaşlarıma teşekkür ederim.

İÇİNDEKİLER

TEŞEKKÜR.....	I
İÇİNDEKİLER.....	II
SİMGELER VE KISALTMALAR LİSTESİ.....	III
ŞEKİLLER LİSTESİ.....	IV
TABLolar LİSTESİ.....	V
ÖZET.....	VI
SUMMARY.....	VII
BÖLÜM 1. GİRİŞ.....	1
BÖLÜM 2. KAYNAK ARAŞTIRMASI.....	3
2.1. BOX-JENKINS MODELLERİNİN KURAMSAL YAPISI.....	3
2.1.1 Durağan Olmayan Doğrusal Olasılıksal Süreçler (ARIMA)...	3
2.1.2 Box-Jenkins Yöntemi.....	4
2.1.2.1 Model Bulma Aşaması.....	5
2.1.2.2 Parametre Tahmini Aşaması.....	7
2.1.2.3 Artık Analizi Aşaması.....	7
2.1.2.4 Gelecek Tahmini Yapma Aşaması.....	9
BÖLÜM 3. MATERYAL VE YÖNTEM.....	10
3.1. PARÇACIK SÜRÜ OPTİMİZASYONU (PSO).....	10
3.2. ARIMA MODELLERİNDE PARAMETRELERİN PSO İLE TAHMİNİ.....	12
BÖLÜM 4. ARAŞTIRMA BULGULARI.....	13
4.1. UYGULAMA.....	13
BÖLÜM 5. TARTIŞMA VE SONUÇ.....	25
KAYNAKLAR.....	26
EKLER.....	28
EK 1- Wei Yöntemi.....	28
EK-2 ARIMA-PSO Yöntemi.....	34
ÖZGEÇMİŞ.....	41

SİMGELER VE KISALTMALAR LİSTESİ

ACF	: Otokorelasyon Fonksiyonu
ADF	: Birim Kök Testi
AIC	: Akaike Bilgi Ölçütü
AR	: Otoregresif Model
ARIMA	: Otoregresif Tümlleşik Hareketli Ortalamalar Modeli
BIC	: Bayesian Bilgi Ölçütü
PACF	: Kısmi Otokorelasyon Fonksiyonu
Y_t	: Gözlenen değerler
Θ	: Toplamsal MA parametreleri
Φ	: Toplamsal AR parametreleri
μ	: Ortalama
δ	: Kesme parametresi
p	: AR sürecinin derecesi
q	: MA sürecinin derecesi
ρ_k	: Otokorelasyon fonksiyonu
H_0	: Yokluk hipotezi
H_1	: Alternatif hipotez

ŞEKİLLER LİSTESİ

Şekil 1. Box-Jenkins Algoritması Akış Diyagramı.....	5
Şekil 2. Parçacık Sürü Optimizasyonu Akış Diyagramı.....	10
Şekil 3. Minimum Sıcaklıklarının Ortalama Veri Grafiği.....	13
Şekil 4. Minimum Sıcaklıklarının Ortalama Otokorelasyon Grafiği.....	14
Şekil 5. Minimum Sıcaklıklarının Ortalama Kısmi Otokorelasyon Grafiği.....	14
Şekil 6. PSO İterasyonlarında Uygunluk Fonksiyonu Değeri.....	17
Şekil 7. Maksimum Sıcaklık Ortalama Veri Grafiği.....	19
Şekil 8. Maksimum Sıcaklık Ortalama Otokorelasyon Grafiği.....	19
Şekil 9. Maksimum Sıcaklık Ortalama Kısmi Otokorelasyon Grafiği.....	20
Şekil 10. PSO İterasyonlarında Uygunluk Fonksiyonu Değeri.....	23

TABLÖLER LİSTESİ

Tablo 1. Bazı ARMA modelleri için otokorelasyon ve kısmi otokorelasyon fonksiyonunun durumları.....	6
Tablo 2. Minimum Sıcaklık Ortalaması ADF Testi.....	15
Tablo 3. Minimum Sıcaklık Ortalaması Wei Yöntemi Parametre Tahminleri.....	16
Tablo 4. Minimum Sıcaklık Ortalaması PSO Parametre Tahminleri.....	16
Tablo 5. Minimum Sıcaklık Ortalaması Öngörü Sonuçları	18
Tablo 6. Maksimum Sıcaklık Ortalaması ADF Testi	21
Tablo 7. Maksimum Sıcaklık Ortalaması Birinci Fark ADF Testi	21
Tablo 8. Maksimum Sıcaklık Ortalaması Parametre Sonuçları	22
Tablo 9. Maksimum Sıcaklık Ortalaması Öngörü Sonuçları	24

ARIMA MODELLERİNDE PARÇACIK SÜRÜ OPTİMİZASYONU İLE PARAMETRE TAHMİNİ

ÖZET

Zaman serilerinin öngörülmesi, gelecek planlamaları açısından oldukça önemlidir. ARIMA modelleri 1970'li yıllardan beri zaman serisi çözümlemesinde en yaygın kullanılan yöntemlerden biridir. ARIMA modelinde parametre tahmini için olabilirlik fonksiyonu maksimizasyonu ya da en küçük kareler fonksiyonunun minimizasyonuna dayalı yöntemler mevcuttur. ARIMA modelinde hareketli ortalama terimlerinin bulunması durumunda model parametrelere göre doğrusal değildir ve bu modelde parametre tahmini için doğrusal olmayan optimizasyon yöntemlerine ihtiyaç vardır. Bu çalışmada ARIMA modelinin parametrelerinin parçacık sürü optimizasyonu ile tahmini için bir algoritma önerilmiştir. Önerilen algoritmanın performansı gerçek hayat zaman serileri üzerinden araştırılmıştır.

Anahtar Kelimeler: ARIMA, Parametre Tahmini, Parçacık Sürü Optimizasyonu, Öngörü.

PARAMETER ESTIMATION IN ARIMA MODEL WITH PARTICLE SWARM OPTIMIZATION

SUMMARY

Forecasting is an important problem for planning future. ARIMA models have been widely used for time series forecasting since 1970. There are maximum likelihood or least square methods to estimate parameters of ARIMA model. The ARIMA model is not linear in terms of its parameters when the model contain moving average terms and this model needs nonlinear optimization techniques for parameter estimation. In this study, a new algorithm is proposed employing particle swarm optimization for parameter estimation in ARIMA models. The performance of the proposed method is compared with some other methods by using real-world time series.

Keywords: ARIMA, Parameter Estimation, Particle Swarm Optimization, Forecasting.

BÖLÜM 1. GİRİŞ

Zaman içerisinde bir olayın gözlemlerinden elde edilen veriler, son derece yaygındır. Ticarete ve ekonomide; haftalık faiz oranları, günlük hisse senedi kapanış fiyatları, aylık fiyat göstergeleri, yıllık satış rakamları ve bunlar gibi birçok şey gözlemlenir. Meteorolojide sıcaklığın günlük artışı ve azalışı, yıllık yağış ve kuraklık belirtileri ve rüzgârın saatteki hızı da aynı şekilde gözlemlenir. Zaman serisi analizinin amacı genellikle iki noktada toplanır: Birincisi gözlenen serinin artışını veren olasılıksal mekanizmayı modellemek ve anlamak, ikincisi ise serinin geçmişine bakarak serinin tahmini veya kestirimidir (Jonathan, 1986). Öngörü tahmini amacıyla kullanılan istatistik yöntemlerinden biri Box ve Jenkins yöntemi diğer adı ile ARIMA olarak bilinen modellerdir. ARIMA modellerinin kullanımı için izlenecek yaklaşım da Box-Jenkins(1976) tarafından ortaya koyulmuştur ve “Box-Jenkins yaklaşımı” olarak adlandırılmıştır. Bu konuyu literatürde, Bates’in erken çalışmalarından bu yana Reid (1968), Granger (1969), ve Clemen (1989) geliştirmiştir, Pai ve Lin (2005) hisse senedi fiyatlarının tahmini için ARIMA modelleri ve destek vektör makinelerine (SVM) dayalı melez bir yaklaşım önermekte, Zhou ve Hu (2008) yılında Grey ve ARIMA modellerini temel alan melez bir yaklaşım önermiştir.

ARIMA modellerinin parametre tahmini için kullanılan birçok yöntem bulunmaktadır. Parçacık sürü optimizasyonu da bu yöntemlerden bir tanesidir. Optimizasyon, muhtemel çözümlerden oluşan bir kümedeki en uygun çözümü, problemle ilgili kısıtları da göz önünde bulundurarak elde etmeye çalışan tekniklere verilen isimdir. Parçacık sürü optimizasyonu (Particle Swarm Optimization), Kennedy ve Eberhart (1995) de geliştirilen popülasyon tabanlı sezgisel bir optimizasyon tekniğidir. Yöntem basitleştirilmiş sosyal modellerin simülasyonu ile geliştirilmiştir. Bu konuyla ilgili literatürde, Genetik algoritma ile ARIMA modelinde parametrelerin tahmini Armano vd. (2005) çalışmasında gerçekleştirilmiştir, Ong vd. (2005) ise ARIMA modelinde model belirleme problemini genetik algoritma ile çözümlmeye çalışmıştır. Behnamian ve Fatemi Ghomi (2010) PSO yöntemini benzetim tavlama (SA)

yöntemiyle melezleyerek zaman serisi öngörü modelinde parametre tahmini için yeni bir meta-sezgisel optimizasyon algoritması ortaya koymuştur. Li-xia, Yi-qi ve Xue-yong (2011), destek vektör makinesi (SVM) yöntemini bir vergi öngörü modeline PSO yöntemiyle birleştiren yeni bir algoritma önermektedir. Asadi vd. (2011) çalışmalarında, ARIMA modelinin parametrelerinin tahmini için PSO yöntemini kullanmıştır. Önerilen modelde, bir zaman serisi verisinin gelecek değeri, geçmiş değerlerinin doğrusal bir fonksiyonu olarak kabul edilir. Önerilen yöntemde veri seti eğitim ve test verisi olarak ayrılarak, test performansı yüksek olacak şekilde parametre tahminleri elde edilmeye çalışılır. Asadi vd. (2011) yılında sabit terimli AR(2) modelini incelemiştir.

Bu çalışmada mevsimsel ve mevsimsel olmayan ARIMA modellerinde, model derecesinin belirlenmesi test kümesine dayalı bir yaklaşımla ve parametrelerin tahmini değiştirilmiş PSO yöntemi ile tahmin edildiği yeni bir algoritma ortaya koyulmuştur. Önerilen yöntem Asadi vd. (2011) yaklaşımı ve literatürdeki diğer yöntemler ile bazı gerçek hayat verileri üzerinden karşılaştırılmıştır. İkinci bölümde ARIMA modelleri ile ilgili genel bilgiler, kuramsal çıkarımlar verilmiştir.

BÖLÜM 2. KAYNAK ARAŞTIRMASI

2.1. BOX-JENKINS MODELLERİNİN KURAMSAL YAPISI

Box-Jenkins modelleri olarak bilinen ARIMA modellerinin zaman serisi öngörü probleminde nasıl uygulanacağı Box ve Jenkins (1976) çalışmasında bir prosedür haline getirilmiştir. Bu bölümde ARIMA modelleri ve Box-Jenkins prosedürü özetlenmiştir.

2.1.1 Durağan Olmayan Doğrusal Olasılıksal Süreçler (ARIMA)

Gerçek hayat zaman serileri durağan olabileceği gibi çoğunlukla durağan olmayan yapıda da olabilmektedir. Durağan zaman serilerinde fark parametresi d sıfır olarak alınırken, bütünleşmenin derecesine göre fark parametresi genelde 1 veya 2 olabilmektedir. Otoregresif terimlerin sayısı p , hareketli ortalama terimlerinin sayısı ise q ile gösterilmektedir. Periyodik olmayan zaman serileri için bu terimlerin sayısı da 1 veya 2 olabilmektedir. ARIMA(p, d, q) modeli (1) eşitliğindeki gibi yazılır.

$$\phi(B)(1-B)^d(Z_t - \mu) = \theta(B)a_t \quad (1)$$

$$\phi(B) = (1 - \phi_1 B - \dots - \phi_p B^p) \quad (2)$$

$$\theta(B) = (1 + \theta_1 B + \dots + \theta_q B^q) \quad (3)$$

Mevsimsel terimleri içeren SARIMA modeli ise Z_t , μ ortalamalı bir zaman serisi için aşağıdaki eşitlikler ile tanımlanır.

$$\phi(B)\Phi(B^s)(1-B)^d(1-B^s)^D(Z_t - \mu) = \theta(B)\Theta(B^s)a_t \quad (4)$$

Burada,

$$\phi(B) = (1 - \phi_1 B - \dots - \phi_p B^p) \quad (5)$$

$$\theta(B) = (1 + \theta_1 B + \dots + \theta_q B^q) \quad (6)$$

$$\Phi(B) = (1 - \Phi_1 B^s - \dots - \Phi_p B^{sP}) \quad (7)$$

$$\Theta(B) = (1 + \Theta_1 B^s + \dots + \Theta_Q B^{sQ}) \quad (8)$$

(4) modeli kısaca SARIMA(p, d, q)(P, D, Q)_s ile gösterilmektedir. SARIMA modellerinin kullanımı için de BJ yaklaşımı kullanılır.

2.1.2 Box-Jenkins Yöntemi

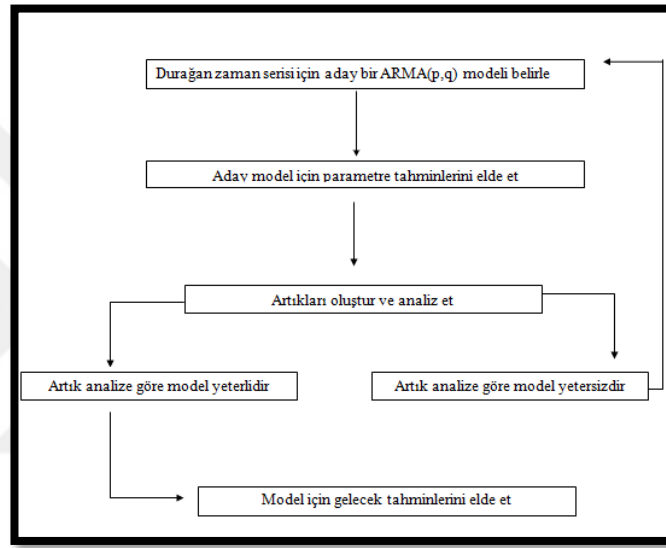
Box-Jenkins yöntemi 1976 yılında George E.P.Box ve Gwilym M. Jenkins tarafından önerilmiştir. Bu yöntem zaman serisine en uygun ARMA modelini bulmayı sağlayan ve günümüze kadar birçok bilimsel çalışmada kullanılmış etkili bir yöntemdir.

Box-Jenkins yöntemi dört ana aşamadan oluşmaktadır. Bunlar,

- Model bulma aşaması
- Parametre tahmini aşaması
- Artık analizi aşaması
- Gelecekle ilgili tahmin yapma aşaması

olarak isimlendirilir.

Box-Jenkins yöntemi ile modellenecek zaman serisinin durağan olması için gerekirse Box-Jenkins yöntemi uygulanmadan önce zaman serisi fark alınarak ya da dönüşümler ile durağanlaştırılmalıdır. Durağan zaman serisi Şekil 1 'deki akış diyagramında verilen şekilde modellenir.



Şekil 1. Box-Jenkins Algoritması Akış Diyagramı.

2.1.2.1 Model Bulma Aşaması

BJ yönteminin uygulanmasında önemli bir aşama model bulma aşamasıdır. Bu kısımda araştırmacı tecrübelerini kullanarak çeşitli grafik ve istatistiklerden yararlanma yoluyla uygun bir model ya da aday modeller seçebilir. Model seçiminde kılavuz olarak teorik bilgilerin özetlendiği Tablo 1 kullanılabilir.

Tablo 1. Bazı ARMA Modelleri için Otokorelasyon ve Kısmi Otokorelasyon Fonksiyonunun Durumları.

Model	Otokorelasyon	K. Otokorelasyon
AR(1)	Üstel ya da sinüsel azalma	Yalnızca ϕ_{11} sıfırdan farklı
ARMA(1,1)	Üstel azalma	Üstel azalma
MA(1)	Yalnızca ρ_1 sıfırdan farklı	Üstel ya da Sinüsel azalma
AR(2)	Üstel ya da sinüsel azalma	Yalnızca r_{11} ve r_{22} sıfırdan farklı
MA(2)	Yalnızca ρ_1 ve ρ_2 sıfırdan farklı	Üstel ya da sinüsel azalma

Tablo 1 içinde gerçek hayat zaman serilerinin çözümlemesinde sık kullanılan AR(1), MA(1), ARMA(1,1), AR(2) ve MA(2) süreçlerinin otokorelasyon ve kısmi otokorelasyon bilgileri verilmiştir. Model bulma aşamasında zaman serisinin otokorelasyon ve kısmi otokorelasyon katsayılarının grafikleri yardımıyla tablodan aday model seçilir. Diğer bir yaklaşım ise değişik p ve q değerleri için Akaike bilgi kriteri gibi ölçütlerin kullanılmasıdır. Bu yöntem bilgisayar uygulamalarında en sık kullanılan yöntemlerden biridir. Genişletilmiş otokorelasyon fonksiyonunda model bulma aşamasında kullanılan bir grafik olarak literatürde yer almaktadır.

2.1.2.2 Parametre Tahmini Aşaması

Parametre tahmin aşamasında ARIMA modellerindeki otoregresif ve hareketli ortalama parametreleri çözümlenen seriye özgü olarak belirlenmeye çalışılır. ARIMA modellerinde parametre tahmini için doğrusal ve doğrusal olmayan en küçük kareler yaklaşımları, en çok olabilirlik yaklaşımı, Bayesci tahmin yöntemleri kullanılmaktadır. Günay ve ark. (2007)'de en çok olabilirlik yaklaşımları aşağıdaki gibi özetlenmiştir.

X_t , $t = 1, 2, \dots, n$ zaman serisi, Z_t 'ler ise standart normal dağılımlı rasgele değişkenlerden oluşan bir beyaz gürültü süreci olsun. ARMA sürecinin logaritmik tam olabilirlik fonksiyonu aşağıdaki gibi yazılabilir:

$$l(\varphi, \theta, \sigma_z^2) = f(\varphi, \theta) - \frac{n}{2} \ln(\sigma_z^2 2\pi) - \frac{S(\varphi, \theta)}{2\sigma_z^2} \quad (9)$$

$f(\varphi, \theta)$, sürecin derecesine göre değişen φ ve θ 'ya dayalı fonksiyon olup, $S(\varphi, \theta)$ ise en küçük kareler fonksiyonudur. Bu fonksiyon,

$$S(\varphi, \theta) = \sum_{t=1}^n Z_t^2 \quad (10)$$

şeklinde verilmektedir. En küçük kareler fonksiyonun hesaplanmasında geriye doğru tahmin yöntemi kullanılabilir.

2.1.2.3 Artık Analizi Aşaması

Bu aşamada Box-Ljung testinden yararlanılabilir. Test istatistiği

$$Q = n(n+2) \sum_{i=1}^K \frac{\rho_i^2}{n-i}, \quad Q \sim \chi_K^2 \quad (11)$$

olmaktadır. Artıklar bağımsız ve normal dağılımlı olduğu sıfır hipotezi aşağıdaki verilmiştir.

$$H_0 : Z_t \sim NID(0, \sigma_z^2) \quad ya \ da \quad Q = 0 \quad (12)$$

α anlam düzeyi için $Q > X_{K, \alpha/2}^2$ ise sıfır hipotezi reddedilir. Yani modelin artıkları uygun değildir sonucuna ulaşılır. Bu durumda model bulma aşamasına geri dönülmesi gerekir. Sıfır hipotezi kabul edilmişse modelin performans ölçütü (AIC, BIC gibi) elde edilir. Ölçüt istenen düzeyde ise gelecekle ilgili tahminler yapılabilir. Eğer istenen düzeyde değilse model bulma aşamasına dönülür. Performans ölçütü en iyi olan model için gelecekle ilgili tahminleri yapılır.

Diğer bir yöntem Partmenteau test istatistiğidir.

$$Q = n \cdot \sum_{k=1}^K \rho_k^2, \quad Q \sim \chi_{k-p-q}^2 \quad (13)$$

Bu testin uygulanması da Box-Ljung testinde olduğu gibidir.

Modeldeki toplam parametre sayısı (m), örneklem büyüklüğü (n), artık kareler toplamı AKT olmak üzere Akaike Bilgi ölçütü (AIC) ve Bayesci Bilgi Ölçütü (BIC) istatistikleri aşağıdaki gibi hesaplanır.

$$AIC(m) = n \log \left(\frac{AKT}{n-m} \right) + 2m \quad (14)$$

$$BIC(m) = n \log \left[\frac{AKT}{n-m} \right] + m \log n \quad (15)$$

2.1.2.4 Gelecek Tahmini Yapma Aşaması

Gelecek tahmini yapma aşamasında modelin öngörülleri elde edilir. ARMA(2,2) modeli için öngörülerin nasıl elde edileceği Günay ve ark. (2007)'de aşağıdaki gibi verilmiştir.

n gözlemlenmiş bir zaman serisi verilsin. $\hat{X}_{n,k}$ ile zaman serisinin k dönem öndeki tahmin değeri (öngörüsü) gösterilsin. Bu değer aşağıdaki gibi elde edilebilir.

$$X_{n+1} = \phi_1 X_n + \phi_2 X_{n-1} + Z_{n+1} + \theta_1 Z_n + \theta_2 Z_{n-1} \quad (16)$$

Burada X_n ve X_{n-1} gözlemlenmiştir. Z_n ve Z_{n-1} yerine de tahmin edilen artıklar kullanılır. Bu durumda tek bilinmeyen Z_{n+1} 'dir. Bu değer de sıfır kabul edilebilir. Bu durumda bir adım öndeki tahmin $\hat{X}_{n,1}$ aşağıdaki gibi hesaplanır.

$$\hat{X}_{n,1} = \hat{\phi}_1 X_n + \hat{\phi}_2 X_{n-1} + \hat{\theta}_1 \hat{Z}_n + \hat{\theta}_2 \hat{Z}_{n-1} \quad (17)$$

$$X_{n+2} = \phi_1 X_{n+1} + \phi_2 X_n + Z_{n+2} + \theta_1 Z_{n+1} + \theta_2 Z_n \quad (18)$$

formülü elde edilir. Burada Z_{n+1} ve Z_{n+2} sıfır olarak alınır. Bilinmeyen X_{n+1} yerine ise $\hat{X}_{n,1}$ alınır ise $\hat{X}_{n,2}$ aşağıdaki gibi hesaplanır.

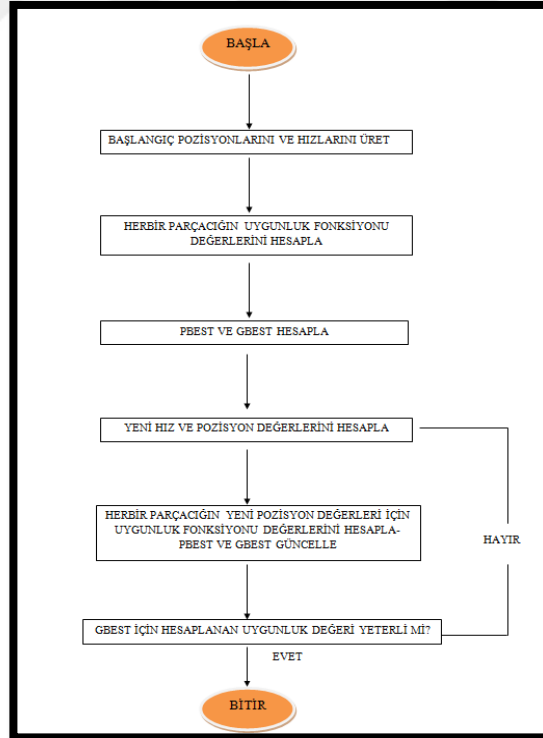
$$\hat{X}_{n,2} = \hat{\phi}_1 \hat{X}_{n,1} + \hat{\phi}_2 X_n + \hat{\theta}_2 \hat{Z}_n \quad (19)$$

Aynı işlemler k-2 kez tekrarlanırsa k dönem ileri tahmin elde edilebilir. k dönem ileri öngörü bulurken, X_{n+j} ($j > 0, j < k$) 'ler tahminleri ile yer değiştirir, Z_{n+j} ($j \geq 0$) 'ler de sıfır kabul edilir. Z_{n-j} ($j \geq 0$) 'lerin yerine ise tahmin edilmiş artıklar getirilir.

BÖLÜM 3. MATERYAL VE YÖNTEM

3.1. PARÇACIK SÜRÜ OPTİMİZASYONU (PSO)

Parçacık sürü optimizasyonu ilk kez Kennedy ve Eberhart (1995) tarafından önerilmiştir. Son yıllarda basitliği, yüksek çözüm kalitesi ve mükemmel yakınsama özellikleri ile yaygın bir uygulama alanı bulmaktadır. PSO yöntemi kuş sürülerinin davranışlarından esinlenerek ortaya koyulmuş sezgisel bir optimizasyon yöntemidir. Standart PSO yönteminde çözümler parçacık olarak isimlendirilir. PSO algoritmasının yakınsama hızı yüksek ve etkin bir algoritması değiştirilmiş parçacık sürü optimizasyonudur. Değiştirilmiş parçacık sürü optimizasyonunun adım adım algoritması aşağıda verilmiştir. Algoritmanın işleyişi ise Şekil 2'deki akış diyagramında görsel hale getirilmiştir.



Şekil 2. Parçacık Sürü Optimizasyonu Akış Diyagramı.

Değiştirilmiş parçacık sürü optimizasyonu

Adım 1. Başlangıç pozisyonlarını ve hızları üret.

$$P_i^{(0)} = \{p_{i,1}^{(0)}, p_{i,2}^{(0)}, \dots, p_{i,d}^{(0)}\}, i = 1, 2, \dots, pn \quad (20)$$

$$V_i^{(0)} = \{v_{i,1}^{(0)}, v_{i,2}^{(0)}, \dots, v_{i,d}^{(0)}\}, i = 1, 2, \dots, pn \quad (21)$$

$P_i^{(0)}$ ve $V_i^{(0)}$ i. parçacığın pozisyon ve hızlarını gösteren vektörlerdir.

Adım 2. Her bir parçacığın başlangıç pozisyon değerlerine göre uygunluk fonksiyonu değerleri f_i ; $i = 1, 2, \dots, pn$ hesaplanır.

Adım 3. $Pbest$ ve $gbest$ belirlenir.

$Pbest$ tüm parçacıkların en iyi pozisyonlarının saklandığı kişisel hafızalarıdır, $gbest$ ise popülasyon için en iyi pozisyonların saklandığı hafızadır.

$$Pbest_i^{(t)} = \{Pb_{i,1}^{(t)}, Pb_{i,2}^{(t)}, \dots, Pb_{i,d}^{(t)}\}, i = 1, 2, \dots, pn \quad (22)$$

$$gbest^{(t)} = \{g_1^{(t)}, g_2^{(t)}, \dots, g_d^{(t)}\} \quad (23)$$

Adım 4. Yeni hız değerleri ve pozisyon değerleri hesaplanır.

$$v_{i,j}^{(t+1)} = w \times v_{i,j}^{(t)} + c_1 \times rand_1 \times (Pb_{i,j}^{(t)} - p_{i,j}^{(t)}) + c_2 \times rand_2 \times (g_j^{(t)} - p_{i,j}^{(t)}) \quad (24)$$

$$p_{i,j}^{(t+1)} = p_{i,j}^{(t)} + v_{i,j}^{(t+1)} \quad (25)$$

Burada $maxt$ ve t sırasıyla maksimum iterasyon sayısı ve geçerli iterasyon sayısını gösterir. c_1 , c_2 ve w ise sırasıyla, bilişsel, sosyal katsayıları ve eylemsizlik ağırlığını gösterir.

Adım 5. Her bir parçacığın yeni pozisyon değerlerine göre uygunluk değerleri yeniden hesaplanır, $Pbest$ ve $gbest$ güncellenir. $gbest$ için hesaplanan uygunluk değeri eğer belirli bir değerin altında ise yada maksimum iterasyon sayısına ulaşıldı ise algoritma durdurulur aksi halde Adım 4'e geri dönülür.

3.2. ARIMA MODELLERİNDE PARAMETRELERİN PSO İLE TAHMİNİ

Mevsimsel ve mevsimsel olmayan ARIMA modellerinde parametre tahmini için doğrusal olmayan fonksiyonların kısıtlı optimizasyonuna ihtiyaç vardır. Bu çalışmada nümerik optimizasyon problemlerinin çözümünde etkin sonuç veren PSO algoritması parametre tahmini için kullanılmıştır. Optimizasyon problemi genel olarak aşağıdaki gibi ifade edilir.

Mevsimsel ARIMA modelindeki parametreler vektörü;

$$\beta = [\mu, \phi_1, \phi_2, \dots, \phi_p, \theta_1, \theta_2, \dots, \theta_q, \Phi_1, \Phi_2, \dots, \Phi_P, \Theta_1, \Theta_2, \dots, \Theta_Q]$$

şeklinde yazılabilir. En küçük kareler fonksiyonu β 'ya göre minimize edilerek modelin parametreleri tahmin edilir. Böylece optimizasyon problemi,

$$\min_{\beta} \sum_{t=\max(p,q,P,Q)}^n a_t^2 \quad (26)$$

olarak ifade edilir. Optimizasyon probleminin çözümünde PSO algoritmasının kullanılması için uygunluk fonksiyonu olarak (26) da verilen en küçük kareler fonksiyonu alınır. Ayrıca bir parçacığın pozisyonları β vektörünün elemanları yani parametrelerin tahmin değerleridir. μ parametresi hariç diğer tüm parametreler -1 ile 1 aralığında sınırlandırılır. Bunun için değiştirilmiş PSO algoritmasındaki her bir pozisyon değeri güncellendikten sonra (27) formülü ile istenilen (-1,1) aralığına çekilir. t. iterasyondaki .parçacığın j. pozisyon değeri için

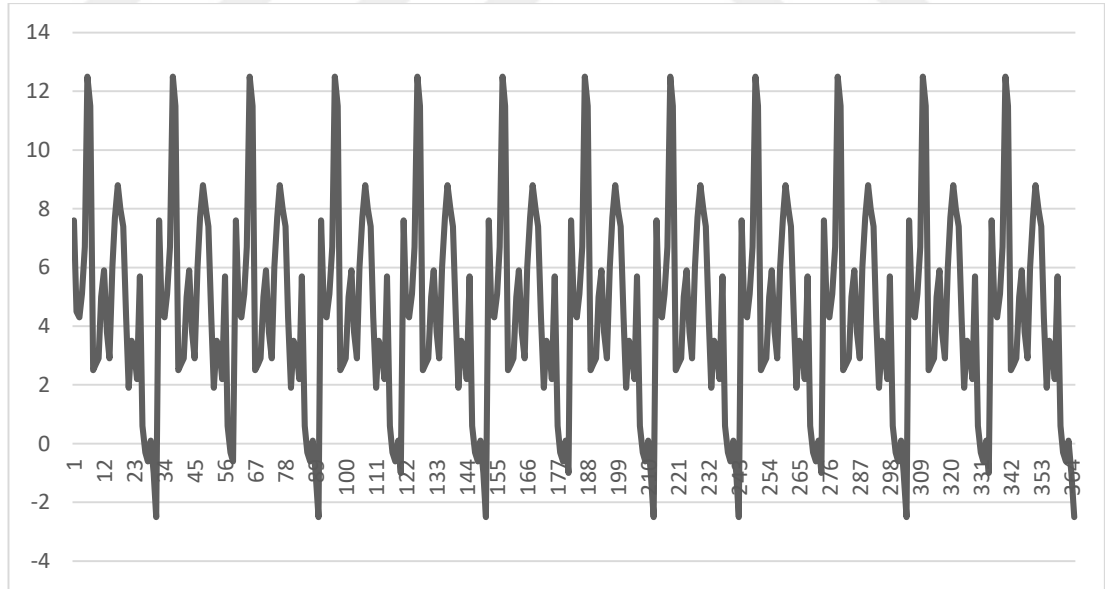
$$p_{i,j}^{(t)} = \max(-1, \min(p_{i,j}^{(t)}, 1)) \quad (27)$$

formülü kullanılır.

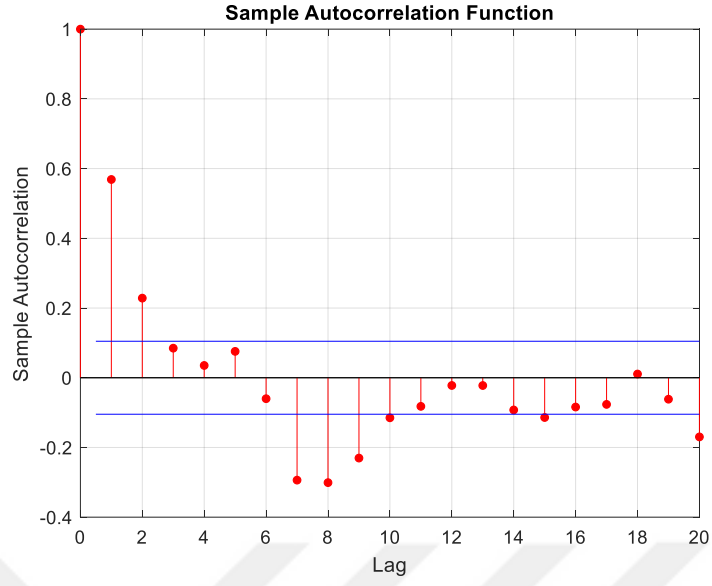
BÖLÜM 4. ARAŞTIRMA BULGULARI

4.1. Uygulama

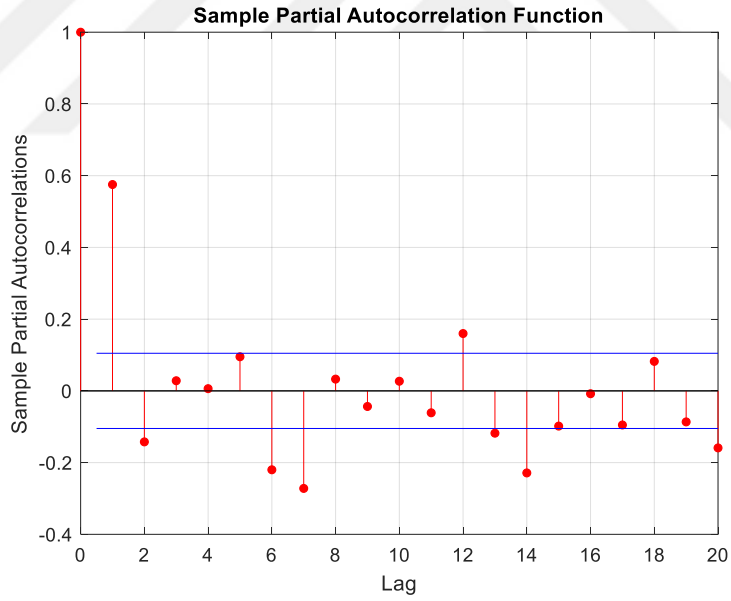
Uygulamada ilk olarak “Giresun Meteoroloji Müdürlüğünden” alınan meteorolojik parametreler ile oluşturulan zaman serileri kullanılarak PSO’ya dayalı ARIMA yönteminin performansı araştırılmıştır. İlk meteorolojik zaman serisi günün yirmi dört saati için Meteoroloji müdürlüğü tarafından ölçülen minimum sıcaklık parametrelerinden alınmıştır. Zaman serisi, günlük zaman serisi olup, gözlemler her günün yirmi dört saatlik minimum sıcaklıklarının ortalaması olarak elde edilmiştir. 1 Ocak 2017 ve 31 Aralık 2017 tarihleri arasındaki günlük zaman serisi 365 gözleme sahip olup, grafiği Şekil 3’de verilmiştir.



Şekil 3. Minimum Sıcaklıklarının Ortalama Veri Grafiği.



Şekil 4. Minimum Sıcaklıklarının Ortalama Otokorelasyon Grafiği.



Şekil 5. Minimum Sıcaklıklarının Ortalama Kısmi Otokorelasyon Grafiği.

Zaman serisindeki bileşenleri daha iyi anlayabilmek için, otokorelasyon ve kısmi otokorelasyon grafikleri elde edilmiş ve Şekil 4 ve Şekil 5’de verilmiştir. Otokorelasyon ve kısmi otokorelasyon grafikleri de seride ilk 8 ay için azalan bir

grafik olduđu, periyodik baskın bir hareketin olmadığını göstermektedir. Serinin durağan bir seri olduđu görölmektedir.

Verideki durağan dışılığın araştırılması için ADF testi uygulanmıştır. Serinin düzeyi için 3 tür ADF testi için elde edilen p-değerleri ve AIC değerleri Tablo 2’de verilmiştir.

Tablo 2. Minimum Sıcaklık Ortalaması ADF Testi.

Model	Gecikme	1
AR	p-değerleri	0,001
	AIC	1833,46
ARD	p-değerleri	0,001
	AIC	1833,46
TS	p-değerleri	0,001
	AIC	1833,46

Tabloya göre p değeri 0,05 den küçük olduđu için seride birim kökün olmadığı ve serinin durağan olduđu sonucuna varılır.

Durağan zaman serisi için ilk olarak Wei (2006) da önerilen birinci türevlere dayalı parametre tahmin yöntemi kullanılarak ARIMA modeli ile zaman serisi çözümlenmiştir. Çözümlemede Matlab’da yazdığımız kod dosyası (Ek 1) kullanılmıştır. Yazılan kodda mevsimsel olmayan veri için 14 adet alternatif model test kümesi performansına göre karşılaştırılarak en iyi model seçilmektedir. Bu 15 model, AR kısmının derecesi 0,1,2 ve 3, MA kısmının derecesi 0,1,2,3 alınarak ve karma modeller de dikkate alınarak tüm kombinasyonlar için toplamda elde edilen 16 modelden AR kısmı ve MA kısmının derecesinin sıfır olduđu durum göz ardı edilerek bulunmaktadır.

Wei'nin yöntemi için en uygun model ARIMA(1,0,0) modeli olup elde edilen parametre tahmin sonuçları Tablo 3'de verilmiştir.

Tablo 3. Minimum Sıcaklık Ortalaması Wei Yöntemi Parametre Tahminleri.

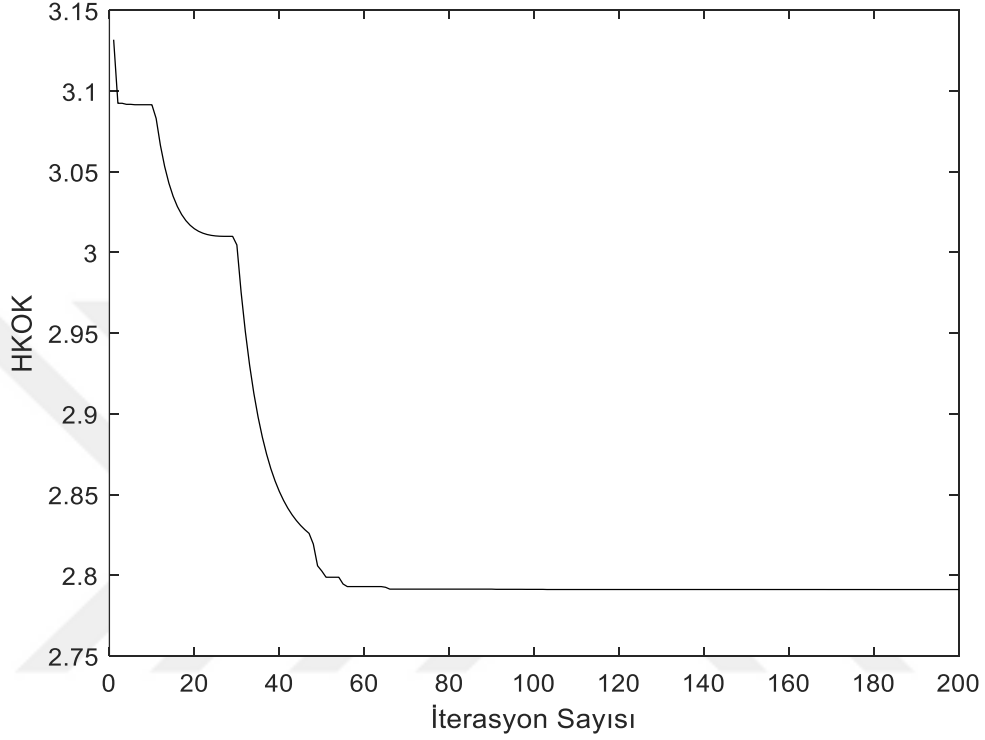
Parametre	Tahmin	t	Anlamlılık Değeri	Kovaryans Matrisi	
Sabit	1,9654	7,9202	0,0000	0,0615	0,0088
AR	0,5617	12,6821	0,0000	-0,0088	0,0020

Durağan zaman serisi ikinci olarak PSO'ya dayalı parametre tahmin yöntemi ile çözümlenmiştir. Çözümlemede Matlab'da yazdığımız kod dosyası (Ek 2) kullanılmıştır. Yazılan kodda mevsimsel olmayan veri için diğer kod ile verilen 15 adet alternatif model test kümesi performansına göre karşılaştırılarak en iyi model seçilmektedir. PSO'ya dayalı yöntemlerde bulunan model ARIMA(1,0,0) modelidir ve parametre tahmin sonuçları Tablo 4 'de verilmiştir.

Tablo 4. Minimum Sıcaklık Ortalaması PSO Parametre Tahminleri.

Parametre	Tahmin	t	Anlamlılık Değeri	Kovaryans Matrisi	
Sabit	1,9654	7,9202	0,0000	0,0615	0,0088
AR	0,5617	12,6821	0,0000	-0,0088	0,0020

Ayrıca ARIMA(1,0,0) modelinde parametre tahmini yapılırken PSO algoritmasında uygunluk fonksiyonunun iterasyonlardaki azalışı Şekil 6’da grafik olarak verilmiştir.



Şekil 6. PSO İterasyonlarında Uygunluk Fonksiyonu Değer Grafiği.

Wei yöntemi ve PSO yöntemlerine dayalı olarak elde edilen öngörü sonuçları Tablo 5’de vermiştir. Tabloda verilen sonuçlar dönüşüm sonrası elde edilen veriler için dir. Tabloda ayrıca Holt (1957) de önerilen Holt’un doğrusal trend yöntemi ve Yapar (2016) da önerilen ATA yöntemlerinden elde edilen öngörü sonuçları da verilmiştir. Öngörüler için hata kareler ortalaması karekök değeri (HKOK) ve hatanın mutlak yüzdelik ortalaması (HMYO) ölçütleri hesaplanmıştır.

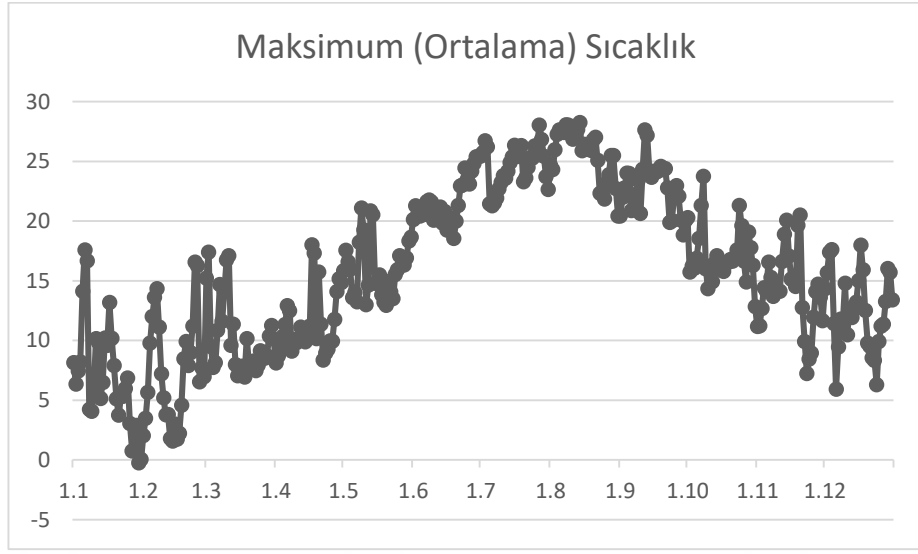
$$HKOK = \sqrt{\frac{1}{n} \sum_{t=1}^n (Z_t - \hat{Z}_t)^2} \quad (28)$$

$$HMYO = \frac{1}{n} \sum_{t=1}^n \left| \frac{Z_t - \hat{Z}_t}{Z_t} \right| \quad (29)$$

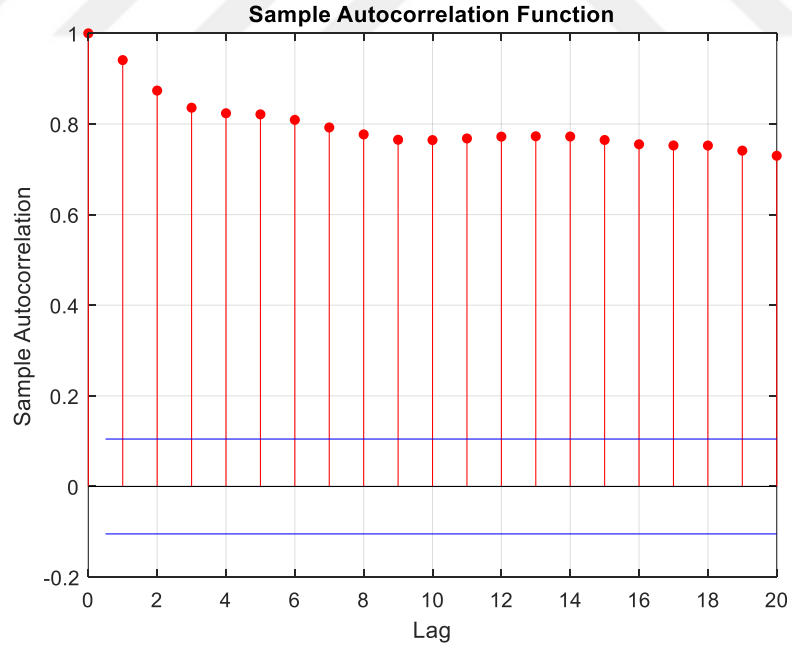
Tablo 5. Minimum Sıcaklık Ortalaması Öngörü Sonuçları.

Test Verisi	Holt Yöntemi	ATA Yöntemi	Wei Yöntemi	ARIMA- PSO
8	8,4131	7,7273	6,9088	6,9128
7,4	8,1153	7,8768	6,4594	6,4619
4,3	7,5886	7,6175	6,1223	6,1237
1,9	5,1130	5,8132	4,3809	4,3766
3,5	2,6947	3,6858	3,0328	3,0240
3,2	3,3196	3,5743	3,9315	3,9258
2,2	3,2440	3,3612	3,7630	3,7567
5,7	2,4684	2,7248	3,2013	3,1931
0,6	4,9308	4,3069	5,1674	5,1657
-0,3	1,6659	2,3152	2,3025	2,2914
-0,6	0,1921	0,9058	1,7969	1,7842
0,1	-0,3928	0,0865	1,6284	1,6151
-1	-0,0046	0,0729	2,0216	2,0096
-2,5	-0,7434	-0,5152	1,4037	1,3897
HKOK	2,0974	2,1536	2,4135	2,4083
HMYO	1,8307	1,6826	2,9292	2,9148

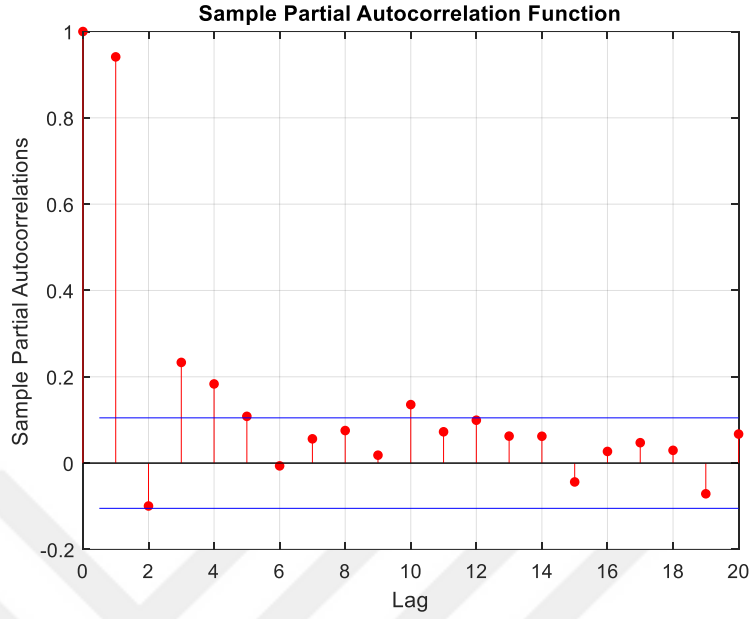
İkinci olarak meteorolojik zaman serisi günün yirmi dört saati için Meteoroloji müdürlüğü tarafından ölçülen maksimum sıcaklık parametrelerinden hesaplanmıştır. Zaman serisi, günlük zaman serisi olup, gözlemler her günün yirmi dört saatlik maksimum sıcaklıklarının ortalaması olarak elde edilmiştir. 1 Ocak 2017 ve 31 Aralık 2017 tarihleri arasındaki günlük zaman serisi 365 gözleme sahip olup, grafiği Şekil 7’de verilmiştir.



Şekil 7. Maksimum Sıcaklık Ortalama Veri Grafiği.



Şekil 8. Maksimum Sıcaklık Ortalama Otokorelasyon Grafiği.



Şekil 9. Maksimum Sıcaklık Ortalama Kısmi Otokorelasyon Grafiği.

Zaman serisindeki bileşenleri daha iyi anlayabilmek için, otokorelasyon ve kısmi otokorelasyon grafikleri elde edilmiş ve Şekil 8 ve Şekil 9’da verilmiştir. Otokorelasyon ve kısmi otokorelasyon grafikleri de seride ilk 8 ay için azalan bir trendin olduğu, ağustos ayından sonra artan bir trendin olduğu ve periyodik baskın bir hareketin olmadığını göstermektedir. Serinin varyansındaki değişimini azaltmak için seriye doğal logaritmik dönüşüm uygulanmıştır. Dönüşümden önce serideki tüm değerlere negatif değerlerden kurtulmak için 5 değeri eklenmiştir. Yani orijinal x_t serisi aşağıdaki dönüşüme tabi tutulmuştur.

$$y_t = \ln(x_t + 5)$$

Verideki durağan dışılığın araştırılması için ADF testi uygulanmıştır. Serinin düzeyi için 3 tür ADF testi için elde edilen p-değerleri ve AIC değerleri Tablo 6’da verilmiştir.

Tablo 6. Maksimum Sıcaklık Ortalaması ADF Testi.

Model	Gecikme	1	2	3	4	5
	p-değerleri	0,527038	0,564656	0,590828	0,559894	0,535896
AR	AIC	-385,895	-400,636	-405,004	-410,499	-407,989
	p-değerleri	0,0011	0,0147	0,0465	0,1000	0,0937
ARD	AIC	-400,501	-409,608	-411,426	-415,086	-412,709
	p-değerleri	0,002056	0,02818	0,101291	0,170633	0,133228
TS	AIC	-402,114	-409,901	-410,974	-414,809	-412,966

Tabloya göre en iyi AIC değerine sahip model “ARD” olup, p değeri 0,05 den büyük olduğu için seride birim kökün olduğu sonucuna varılır. Serinin birinci farkı alınarak ADF testi tekrar uygulanırsa aşağıdaki Tablo 7’de elde edilir.

Tablo 7. Maksimum Sıcaklık Ortalaması Birinci Fark ADF Testi.

Model	Gecikme	1	2	3	4	5
	p-values	0,00	0,00	0,00	0,00	0,00
AR	AIC	-402,586	-406,986	-412,448	-409,904	-406,862
	p-values	0,0010	0,0010	0,0010	0,0010	0,0010
ARD	AIC	-400,607	-405,01	-410,439	-407,889	-404,844
	p-values	0,0010	0,0010	0,0010	0,0010	0,0010
TS	AIC	-398,729	-403,164	-408,466	-405,878	-402,83

Tablo 7 incelenirse en iyi modelin 5 gecikmede ‘AR’ modeli olduğu anlaşılmaktadır. ADF testi sonucuna göre ise p değeri 0,05 den küçük olduğu için seri birinci farkında durağan olmaktadır.

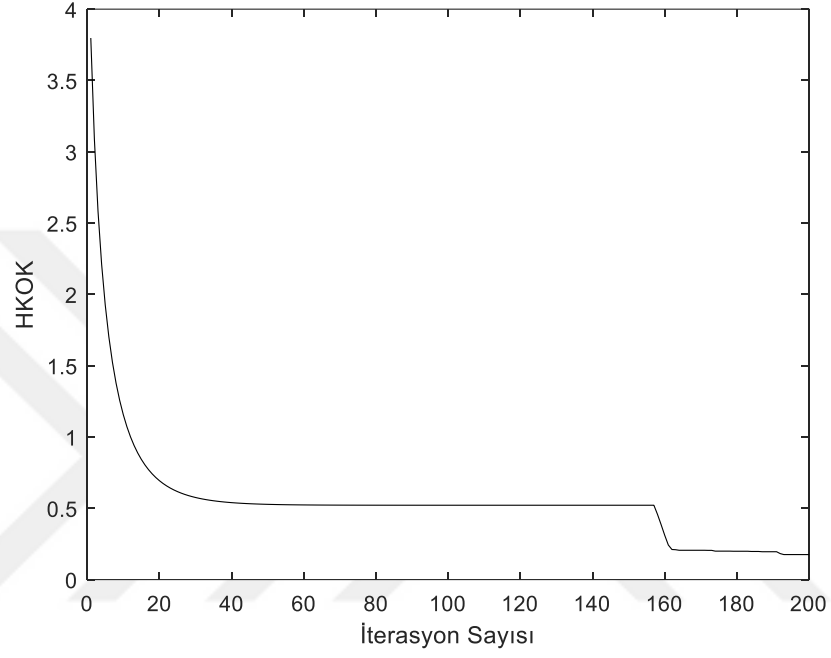
Durağan zaman serisi için ilk olarak Wei (2006) da önerilen birinci türevlere dayalı parametre tahmin yöntemi kullanılarak ARIMA modeli ile zaman serisi çözümlenmiştir. Çözümlemede Matlab’da yazdığımız kod dosyası (EK 1) kullanılmıştır. Yazılan kodda mevsimsel olmayan veri için 15 adet alternatif model test kümesi performansına göre karşılaştırılarak en iyi model seçilmektedir. Bu 15 model, AR kısmının derecesi 0,1,2 ve 3, MA kısmının derecesi 0,1,2,3 alınarak ve karma modeller de dikkate alınarak tüm kombinasyonlar için toplamda elde edilen 16 modelden AR kısmı ve MA kısmının derecesinin sıfır olduğu durum göz ardı edilerek bulunmaktadır. Wei (2006) yöntemi için en uygun model ARIMA(0,1,0) modeli olarak elde edilmektedir.

Durağan zaman serisi ikinci olarak PSO’ya dayalı parametre tahmin yöntemi ile çözümlenmiştir. Çözümlemede Matlab’da yazdığımız kod dosyası (Ek 2) kullanılmıştır. Yazılan kodda mevsimsel olmayan veri için diğer kod ile verilen 15 adet alternatif model test kümesi performansına göre karşılaştırılarak en iyi model seçilmektedir. PSO’ya dayalı yöntemlerde bulunan model ARIMA(1,1,1) modelidir ve parametre tahmin sonuçları Tablo 8’de verilmiştir.

Tablo 8. Maksimum Sıcaklık Ortalaması Parametre Sonuçları.

Parametre	Tahmin	T	Anlamlılık Değeri	Kovaryans Matrisi		
Sabit	0,00	-0,32	0,75	0,0001	0,0004	-0,0005
AR1	0,58	3,37	0,00	0,0004	0,0295	-0,0284
MA1	-0,81	-4,65	0,00	-0,0005	-0,0284	0,0302

Ayrıca ARIMA(1,1,1) modelinde parametre tahmini yapılırken PSO algoritmasında uygunluk fonksiyonunun iterasyonlardaki azalışı Şekil 10’da grafik olarak verilmiştir.



Şekil 10. PSO İterasyonlarında Uygunluk Fonksiyonu Değeri.

Wei (2006) ve PSO yöntemlerine dayalı olarak elde edilen öngörü sonuçları Tablo 9’da verilmiştir. Tabloda verile sonuçlar dönüşüm sonrası elde edilen veriler için elde edilmiştir. Tabloda ayrıca Holt ve ATA yöntemlerinden elde edilen öngörü sonuçları da verilmiştir.

Tablo 9. Maksimum Sıcaklık Ortalaması Öngörü Sonuçları.

Test Verisi	Holt Yöntemi	ATA Yöntemi	Wei Yöntemi	ARIMA-PSO
15,9500	18,0530	17,8534	18,0125	16,4173
12,5208	15,9869	16,0921	15,9500	15,14789
9,8083	12,5525	12,7484	12,5208	12,6423
9,5167	9,8336	9,9962	9,8083	10,4375
8,5667	9,5414	9,5544	9,5167	10,0443
8,3500	8,5908	8,6371	8,5667	9,1469
6,3000	8,3738	8,3724	8,3500	8,8266
9,9375	6,3203	6,4455	6,3000	7,0468
11,2250	9,9632	9,6245	9,9375	9,7198
11,3792	11,2533	11,0948	11,2250	10,6927
13,2708	11,4085	11,3642	11,3792	10,8695
16,0458	13,3018	13,1088	13,2708	12,3258
15,7125	16,0815	15,7864	16,0458	14,4704
13,4125	15,7490	15,7448	15,7125	14,4724
HKOK	2,0816	2,1238	0,1290	0,1286
HMYO	0,1564	0,1613	0,0379	0,0408

BÖLÜM 5. TARTIŞMA VE SONUÇ

Bu çalışmada ARIMA modellerinde parametre tahmini aşamasında parçacık sürü optimizasyonunun kullanılması tartışılmıştır. Wei (2006) çalışmasında önerilen klasik birinci türevlere dayalı parametre tahmin yöntemi ile karşılaştırmalı sonuçlar elde edilmiştir. Giresun Meteoroloji Müdürlüğünden alınan meteorolojik parametreler ile oluşturulan Minimum ve Maksimum sıcaklık verileri üzerine yapılan uygulamada Wei (2006) yöntemi ile PSO'ya dayalı ARIMA yöntemlerine ek olarak HOLT ve ATA yöntemleri de karşılaştırılmıştır.

Minimum sıcaklık verisi için Wei (2006) yöntemine göre elde edilen en uygun model ARIMA(1,0,0) modeli olup, PSO'ya dayalı yöntemde bulunan model ARIMA(1,0,0) modelidir. RMSE değeri '2,4083' olan PSO'ya dayalı yöntem Wei(2006) yöntemine göre daha iyi sonuç vermiş ve en iyi performansa sahip model PSO'ya dayalı ARIMA yöntemi olan ARIMA(1,0,0) modelidir. Ayrıca karşılaştırdığımız diğer yöntemlerden biri olan HOLT yöntemi ise '2,0974' olan RMSE değeri ile veri setine en uygun sonuç veren yöntem olmuştur.

Maksimum sıcaklık verisi için Wei (2006) yöntemi için en uygun model ARIMA(0,1,0) modeli olup, PSO'ya dayalı yöntemlerde bulunan model ARIMA(1,1,1) modelidir. RMSE değeri '0,1286' olan PSO'ya dayalı yöntem Wei(2006) yöntemi dahil diğer tüm yöntemlere göre daha iyi sonuç vermiş ve en iyi performansa sahip model PSO'ya dayalı ARIMA yöntemi olan ARIMA(1,1,1) modelidir.

KAYNAKLAR

- Asadi S., Tavakoli A., Hejaz S.R. 2011. A New Hybrid for Improvement of Auto-Regressive Integrated Moving Average Models Applying Particle Swarm Optimization, Expert Systems with Applications: An International Journal, Tarrytown NY USA, 39, 5332-5337.
- Bates J. M., Granger C. W. J. 1969. The Combination of Forecasts. Operational Research Society.
- Behnamian J, Ghomi SMTF. 2010. Development of a PSO–SA hybrid Metaheuristic For A New Comprehensive Regression Model to Time-Series Forecasting, Expert Systems With Applications, 37, 974-984.
- Box G.E.P., Jenkins G.M. 1976. Time Series Analysis: Forecasting and Control, University of Wisconsin U.S.A and University of Lancaster U.K.
- Cura T. 2008. Modern Sezgisel Teknikler ve Uygulamaları, 1. baskı, Papatya Yayıncılık Eğitim, İstanbul.
- Çelenli A.Z. 2013. Parçacık Sürü Optimizasyonuna Dayalı Portföy Optimizasyonu, Ondokuz Mayıs Üniversitesi Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Samsun.
- Duru Ö. 2007. Zaman Serilerinde Analizinde ARIMA Modelleri ve Bir Uygulama, İstanbul Üniversitesi Sosyal Bilimler Enstitüsü, Yüksek Lisans Tezi, İstanbul.
- Erdoğan E. 2006. Zaman Serilerinde ARIMA Modelleri, Muğla Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Muğla.
- Günay S., Eğrioglu E., Aladağ Ç.H. 2012. Tek Değişkenli Zaman Serilerine Giriş, Hacettepe Üniversitesi Yayınları, Ankara.
- Holt C. C. 1957. Forecasting seasonals and trends by exponentially weighted averages. O.N.R. Memorandum 52/1957, Carnegie Institute of Technology. Reprinted with discussion in 2004. International Journal of Forecasting, 20, 5 – 13.
- Kennedy J., Eberhart R. 1995. Particle Swarm Optimization, Proceedings of IEEE International Conference on Neural Networks, Piscataway.
- Liu Li-xia, Zhuang Yi-qi and Xue-yong Liu. 2011. Tax Forecasting Theory and Model Based on SVM Optimized by PSO, Expert Systems with Applications: An International Journal, Tarrytown NY USA, 38, 116-120.

Pai P.F., Lin C.S. 2004. A hybrid ARIMA and Support Vector Machines Model in Stock Price Forecasting, *Omega*, 33, 497-505.

Wei W.W.S. 2006. *Time Series Analysis Univariate and Multivariate Methods*, Second Edition, Temple University.

Yapar G. 2018. Modified simple exponential smoothing, *Hacettepe Journal of Mathematics and Statistics*, 47(3), 741-754.



EKLER

EK 1- Wei Yöntemi

```
function
[RMSEtest,MAPEtest,xtahg,xtahtestg,Alfat,Valfa,thesap,Sig]=armapqtahtestMs(x,M
p,d,D,s,Mq,ntest)
%Derece matris te veriliyor MP=[1 4] , Mq=[1 2 4 8]
p=max(Mp);q=max(Mq);np=length(Mp);nq=length(Mq);
rand('seed',123456789);
if Mq==0
    nq=0;
end
if Mp==0
    np=0;
end
if (d==1)&&(D==0)
    y=x;
    x=diff(x);
elseif (d==1)&&(D==1)
    y=x;
    x=diff(x);
    nx=length(x);
    x=x(s+1:nx)-x(1:nx-s);
elseif (d==0)&&(D==1)
    y=x;
    nx=length(x);
    x=x(s+1:nx)-x(1:nx-s);
elseif (d==0)&&(D==0)
    y=x;
end
ny=length(y);
md=max(p,q);n=length(x)-ntest;
Alfa0=unifrnd(0,1,np+nq+1,1);
if (q>0)&&(p>0)
    hata=10;eps0=10^-4;
    k=0;
    while hata>eps0
        k=k+1
        if k>100
            break
        end
        for j=1:md
            e(j)=0;
```

```

        xtah(j)=x(j);
    end
    for i=md+1:n
        xtah(i)=Alfa0(1);
        for j=1:np
            xtah(i)=xtah(i)+Alfa0(j+1)*x(i-Mp(j));
        end
        for j=1:nq
            xtah(i)=xtah(i)+Alfa0(j+np+1)*e(i-Mq(j));
        end
        e(i)=x(i)-xtah(i);
    end
    Xa=[ones(n,1),lagmatrix(x(1:n),Mp),lagmatrix(e',Mq)];
    Xa=Xa(md+1:n,:);
    Delta=inv(Xa'*Xa)*Xa'*e(md+1:n)';
    Alfat=Alfa0+Delta;
    hata=norm(Alfat-Alfa0,2);
    Alfa0=Alfat;
end
Sigmatah=mean(e(md+1:n).^2);
Valfa=Sigmatah*inv(Xa'*Xa);
for j=1:np+nq+1
    thesap(j)=Alfat(j)/(Valfa(j,j)^0.5);
    if thesap(j)>0
        Sig(j)=2*(1-tcdf(thesap(j),n-1));
    else
        Sig(j)=2*(tcdf(thesap(j),n-1));
    end
end
elseif (p>0)&&(q==0)
    Xa=[ones(n,1),lagmatrix(x(1:n),Mp)];
    Xa=Xa(md+1:n,:);
    Alfat=inv(Xa'*Xa)*Xa'*x(md+1:n);
    e=x(md+1:n)-Xa*Alfat;
    Sigmatah=mean(e.^2);
    Valfa=Sigmatah*inv(Xa'*Xa);
    for j=1:np+nq+1
        thesap(j)=Alfat(j)/(Valfa(j,j)^0.5);
        if thesap(j)>0
            Sig(j)=2*(1-tcdf(thesap(j),n-1));
        else
            Sig(j)=2*(tcdf(thesap(j),n-1));
        end
    end
elseif (p==0)&&(q>0)
    hata=10;eps0=10^-4;
    k=0;
    while hata>eps0

```

```

k=k+1
if k>100
    break
end
for j=1:md
    e(j)=0;
    xtah(j)=x(j);
end
for i=md+1:n
    xtah(i)=Alfa0(1);
    for j=1:nq
        xtah(i)=xtah(i)+Alfa0(j+np+1)*e(i-Mq(j));
    end
    e(i)=x(i)-xtah(i);
end
Xa=[ones(n,1),lagmatrix(e',Mq)];
Xa=Xa(md+1:n,:);
Delta=inv(Xa'*Xa)*Xa'*e(md+1:n)';
Alfat=Alfa0+Delta;
hata=norm(Alfat-Alfa0,2);
Alfa0=Alfat;
end
Sigmatah=mean(e(md+1:n).^2);
Valfa=Sigmatah*inv(Xa'*Xa);
for j=1:np+nq+1
    thesap(j)=Alfat(j)/(Valfa(j,j)^0.5);
    if thesap(j)>0
        Sig(j)=2*(1-tcdf(thesap(j),n-1));
    else
        Sig(j)=2*(tcdf(thesap(j),n-1));
    end
end
end
%Test Kümesi Öngörüler
if (d==1)&&(D==0)
if q>0
    for j=1:md
        e(j)=0;
        e2(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n+ntest
        xtah(i)=Alfat(1);
        for j=1:np
            xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
        end
        for j=1:nq
            xtah(i)=xtah(i)+Alfat(j+np+1)*e(i-Mq(j));

```

```

    end
    e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
end
xtahg=xtah(md+1:n+ntest)'+y(md+1:ny-1);
er=y(md+2:ny)-xtahg;
er2=abs((y(md+2:ny)-xtahg)./y(md+2:ny));
else
    for j=1:md
        e(j)=0;
        e2(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n+ntest
        xtah(i)=Alfat(1);
        for j=1:np
            xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
        end
        e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
    end
    xtahg=xtah(md+1:n+ntest)'+y(md+1:ny-1);
    er=y(md+2:ny)-xtahg;
    er2=abs((y(md+2:ny)-xtahg)./y(md+2:ny));
end
ner=length(er);
RMSEtest=mean(er(ner-ntest+1:ner).^2)^0.5;
MAPEtest=mean(er2(ner-ntest+1:ner));
elseif (d==1)&&(D==1)
    if q>0
        for j=1:md
            e(j)=0;
            e2(j)=0;
            xtah(j)=x(j);
        end
        for i=md+1:n+ntest
            xtah(i)=Alfat(1);
            for j=1:np
                xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
            end
            for j=1:nq
                xtah(i)=xtah(i)+Alfat(j+np+1)*e(i-Mq(j));
            end
            e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
        end
        xtahg=xtah(md+1:n+ntest)'+y(md+s+1:ny-1)+y(md+2:ny-s)-y(md+1:ny-s-1);
        er=y(s+md+2:ny)-xtahg;er2=abs((y(s+md+2:ny)-xtahg)./y(s+md+2:ny));
    else
        for j=1:md
            e(j)=0;

```

```

e2(j)=0;
xtah(j)=x(j);
end
for i=md+1:n+ntest
    xtah(i)=Alfat(1);
    for j=1:np
        xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
    end
    e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
end
xtahg=xtah(md+1:n+ntest)+y(md+s+1:ny-1)+y(md+2:ny-s)-y(md+1:ny-s-1);
er=y(s+md+2:ny)-xtahg;er2=abs((y(s+md+2:ny)-xtahg)./y(s+md+2:ny));
end
ner=length(er);
RMSEtest=mean(er(ner-ntest+1:ner).^2)^0.5;
MAPEtest=mean(er2(ner-ntest+1:ner));
elseif (d==0)&&(D==1)
if q>0
    for j=1:md
        e(j)=0;
        e2(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n+ntest
        xtah(i)=Alfat(1);
        for j=1:np
            xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
        end
        for j=1:nq
            xtah(i)=xtah(i)+Alfat(j+np+1)*e(i-Mq(j));
        end
        e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
    end
    xtahg=xtah(md+1:n+ntest)+y(md+1:ny-s);
    er=y(s+md+1:ny)-xtahg;er2=abs((y(s+md+1:ny)-xtahg)./y(s+md+1:ny));
else
    for j=1:md
        e(j)=0;
        e2(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n+ntest
        xtah(i)=Alfat(1);
        for j=1:np
            xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
        end
        e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
    end
end

```

```

    xtahg=xtah(md+1:n+ntest)+y(md+1:ny-s);
    er=y(s+md+1:ny)-xtahg;er2=abs((y(s+md+1:ny)-xtahg)./y(s+md+1:ny));
end
ner=length(er);
RMSEtest=mean(er(ner-ntest+1:ner).^2)^0.5;
MAPEtest=mean(er2(ner-ntest+1:ner));
elseif (d==0)&&(D==0)
if q>0
    for j=1:md
        e(j)=0;
        e2(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n+ntest
        xtah(i)=Alfat(1);
        for j=1:np
            xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
        end
        for j=1:nq
            xtah(i)=xtah(i)+Alfat(j+np+1)*e(i-Mq(j));
        end
        e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
    end
else
    for j=1:md
        e(j)=0;
        e2(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n+ntest
        xtah(i)=Alfat(1);
        for j=1:np
            xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
        end
        e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
    end
end
ner=length(xtah);
RMSEtest=mean(e(n:n+ntest).^2)^0.5;
MAPEtest=mean(e2(n:n+ntest));
xtahg=xtah;
xtahtestg=xtahg(ner-ntest+1:ner);
end
xtahtestg=xtahg(ner-ntest+1:ner);

end

```

EK-2 ARIMA-PSO Yöntemi

```
function
[RMSEtest,MAPEtest,xtahg,xtahtestg,gbest,gbestf,rn,Sig,thesap,Valfa,Alfat]=armap
dqspsoisig(x,Mp,d,D,s,Mq,ntest)
% HKTARMA(Alfa0,x,Mp,d,D,s,Mq,ntest)
% x-zaman serisi
% ps-parçacık sayısı
% c1- Bilişsel katsayı
% c2- Sosyal katsayı
% w- Eylemsizlik ağırlığı
% maxitr- Maksimum iterasyon sayısıdır.
% P- Parçacıkların pozisyon değerleri
% V- Parçacıkların hız değerleri
% Pbest, gbest
% Pbestf, gbestf, Pf - Uygunluk değerleri vektörler
%Başlangıç hızlarını U(0,1) dağılımından rastgele üretelim.
saat=clock;
%rand('seed',saat(6)*100000000);
rand('seed',670300000);
rn=rand('seed');
ps=30;c1=2;c2=2;w=0.9;maxitr=200;
p=max(Mp);q=max(Mq);
np=length(Mp);nq=length(Mq);
if Mq==0
    nq=0;
end
if Mp==0
    np=0;
end
end
if (d==1)&&(D==0)
    y=x;
    x=diff(x);
elseif (d==1)&&(D==1)
    y=x;
    x=diff(x);
    nx=length(x);
    x=x(s+1:nx)-x(1:nx-s);
elseif (d==0)&&(D==1)
    y=x;
    nx=length(x);
    x=x(s+1:nx)-x(1:nx-s);
elseif (d==0)&&(D==0)
    y=x;
end
```

```

ny=length(y);
md=max(p,q);n=length(x)-ntest;
pd=np+nq+1;
V=unifrnd(0,1,ps,pd);
%Başlangıç pozisyonlarını üretelim.
P1=unifrnd(-1,1,ps,pd-1);
P2=unifrnd(-100,100,ps,1);
P=[P2,P1];
%Uygunluk Değerleri hesaplanır.
for i=1:ps
    Pf(i)=HKTARMA(P(i,:),x,Mp,d,D,s,Mq,ntest);
end
%Pbest ve gbest oluşturulur.
Pbest=P;
Pbestf=Pf;
ms=find(Pbestf==min(Pbestf));
ms=ms(1);
%Minimum uygunluk değerine sahip parçacık numarası ms dir.
gbest=Pbest(ms,:);
gbestf=Pbestf(ms);
%İterasyonlar başlasın
kt=0;
for i=1:maxitr
    kt=kt+1;
    %Hızlar ve pozisyonlar güncellensin.
    if kt>500
        P1=unifrnd(-1,1,ps,pd-1);
        kt=0;
    end
    for j1=1:ps
        for j2=1:pd
            V(j1,j2)=w*V(j1,j2)+2*c1*unifrnd(0,1)*(Pbest(j1,j2)-
P(j1,j2))+2*c2*unifrnd(0,1)*(gbest(j2)-P(j1,j2));
            V(j1,j2)=max(min(V(j1,j2),0.1),-0.1);
            P(j1,j2)=P(j1,j2)+V(j1,j2);
            if j2>=2
                P(j1,j2)=max(min(P(j1,j2),0.99),-0.99);
            end
        end
        end
        Pf(j1)=HKTARMA(P(j1,:),x,Mp,d,D,s,Mq,ntest);
        if Pf(j1)<Pbestf(j1)
            Pbest(j1,:)=P(j1,:);
            Pbestf(j1)=Pf(j1);
        end
    end
    %gbest güncellensin
    ms=find(Pbestf==min(Pbestf));
    ms=ms(1);

```

```

gbest=Pbest(ms,:);
gbestf=Pbestf(ms);
GB(i)=gbestf;
end
plot(GB,'color','black')
xlabel('İterasyon Sayısı')
ylabel('HKOK')
%Test Kısımı
Alfat=gbest;
if (d==1)&&(D==0)
if q>0
for j=1:md
e(j)=0;
e2(j)=0;
xtah(j)=x(j);
end
for i=md+1:n+ntest
xtah(i)=Alfat(1);
for j=1:np
xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
end
for j=1:nq
xtah(i)=xtah(i)+Alfat(j+np+1)*e(i-Mq(j));
end
e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
end
xtahg=xtah(md+1:n+ntest)+y(md+1:ny-1);
er=y(md+2:ny)-xtahg;
er2=abs((y(md+2:ny)-xtahg)./y(md+2:ny));
else
for j=1:md
e(j)=0;
e2(j)=0;
xtah(j)=x(j);
end
for i=md+1:n+ntest
xtah(i)=Alfat(1);
for j=1:np
xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
end
e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
end
xtahg=xtah(md+1:n+ntest)+y(md+1:ny-1);
er=y(md+2:ny)-xtahg;
er2=abs((y(md+2:ny)-xtahg)./y(md+2:ny));
end
ner=length(er);
RMSEtest=mean(er(ner-ntest+1:ner).^2)^0.5;

```

```

MAPEtest=mean(er2(ner-ntest+1:ner));
elseif (d==1)&&(D==1)
if q>0
    for j=1:md
        e(j)=0;
        e2(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n+ntest
        xtah(i)=Alfat(1);
        for j=1:np
            xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
        end
        for j=1:nq
            xtah(i)=xtah(i)+Alfat(j+np+1)*e(i-Mq(j));
        end
        e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
    end
    xtahg=xtah(md+1:n+ntest)+y(md+s+1:ny-1)+y(md+2:ny-s)-y(md+1:ny-s-1);
    er=y(s+md+2:ny)-xtahg;er2=abs((y(s+md+2:ny)-xtahg)./y(s+md+2:ny));
else
    for j=1:md
        e(j)=0;
        e2(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n+ntest
        xtah(i)=Alfat(1);
        for j=1:np
            xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
        end
        e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
    end
    xtahg=xtah(md+1:n+ntest)+y(md+s+1:ny-1)+y(md+2:ny-s)-y(md+1:ny-s-1);
    er=y(s+md+2:ny)-xtahg;er2=abs((y(s+md+2:ny)-xtahg)./y(s+md+2:ny));
end
ner=length(er);
RMSEtest=mean(er(ner-ntest+1:ner).^2)^0.5;
MAPEtest=mean(er2(ner-ntest+1:ner));
elseif (d==0)&&(D==1)
if q>0
    for j=1:md
        e(j)=0;
        e2(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n+ntest
        xtah(i)=Alfat(1);

```

```

    for j=1:np
        xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
    end
    for j=1:nq
        xtah(i)=xtah(i)+Alfat(j+np+1)*e(i-Mq(j));
    end
    e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
end
xtahg=xtah(md+1:n+ntest)+y(md+1:ny-s);
er=y(s+md+1:ny)-xtahg;er2=abs((y(s+md+1:ny)-xtahg)./y(s+md+1:ny));
else
    for j=1:md
        e(j)=0;
        e2(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n+ntest
        xtah(i)=Alfat(1);
        for j=1:np
            xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
        end
        e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
    end
    xtahg=xtah(md+1:n+ntest)+y(md+1:ny-s);
    er=y(s+md+1:ny)-xtahg;er2=abs((y(s+md+1:ny)-xtahg)./y(s+md+1:ny));
end
ner=length(er);
RMSEtest=mean(er(ner-ntest+1:ner).^2)^0.5;
MAPEtest=mean(er2(ner-ntest+1:ner));
elseif (d==0)&&(D==0)
if q>0
    for j=1:md
        e(j)=0;
        e2(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n+ntest
        xtah(i)=Alfat(1);
        for j=1:np
            xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
        end
        for j=1:nq
            xtah(i)=xtah(i)+Alfat(j+np+1)*e(i-Mq(j));
        end
        e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
    end
else
    for j=1:md

```

```

e(j)=0;
e2(j)=0;
xtah(j)=x(j);
end
for i=md+1:n+nstest
    xtah(i)=Alfat(1);
    for j=1:np
        xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
    end
    e(i)=x(i)-xtah(i); e2(i)=abs((x(i)-xtah(i))/x(i));
end
end
ner=length(xtah);
RMSEtest=mean(e(n:n+nstest).^2)^0.5;
MAPEtest=mean(e2(n:n+nstest));
xtahg=xtah;
xtahstestg=xtahg(ner-nstest+1:ner);
end
xtahstestg=xtahg(ner-nstest+1:ner);
% Sig Hesabi
n=length(x);
if (q>0)&&(p>0)
    for j=1:md
        e(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n
        xtah(i)=Alfat(1);
        for j=1:np
            xtah(i)=xtah(i)+Alfat(j+1)*x(i-Mp(j));
        end
        for j=1:nq
            xtah(i)=xtah(i)+Alfat(j+np+1)*e(i-Mq(j));
        end
        e(i)=x(i)-xtah(i);
    end
    Xa=[ones(n,1),lagmatrix(x(1:n),Mp),lagmatrix(e',Mq)];
    Xa=Xa(md+1:n,:);
    Sigmatah=mean(e(md+1:n).^2);
    Valfa=Sigmatah*inv(Xa'*Xa);
    for j=1:np+nq+1
        thesap(j)=Alfat(j)/(Valfa(j,j)^0.5);
        if thesap(j)>0
            Sig(j)=2*(1-tcdf(thesap(j),n-1));
        else
            Sig(j)=2*(tcdf(thesap(j),n-1));
        end
    end
end

```

```

elseif (p>0)&&(q==0)
    Xa=[ones(n,1),lagmatrix(x(1:n),Mp)];
    Xa=Xa(md+1:n,:);
    e=x(md+1:n)-Xa*Alfat';
    Sigmatah=mean(e.^2);
    Valfa=Sigmatah*inv(Xa'*Xa);
    for j=1:np+nq+1
        thesap(j)=Alfat(j)/(Valfa(j,j)^0.5);
        if thesap(j)>0
            Sig(j)=2*(1-tcdf(thesap(j),n-1));
        else
            Sig(j)=2*(tcdf(thesap(j),n-1));
        end
    end
elseif (p==0)&&(q>0)
    for j=1:md
        e(j)=0;
        xtah(j)=x(j);
    end
    for i=md+1:n
        xtah(i)=Alfat(1);
        for j=1:nq
            xtah(i)=xtah(i)+Alfat(j+np+1)*e(i-Mq(j));
        end
        e(i)=x(i)-xtah(i);
    end
    Xa=[ones(n,1),lagmatrix(e',Mq)];
    Xa=Xa(md+1:n,:);
    Sigmatah=mean(e(md+1:n).^2);
    Valfa=Sigmatah*inv(Xa'*Xa);
    for j=1:np+nq+1
        thesap(j)=Alfat(j)/(Valfa(j,j)^0.5);
        if thesap(j)>0
            Sig(j)=2*(1-tcdf(thesap(j),n-1));
        else
            Sig(j)=2*(tcdf(thesap(j),n-1))
        end ;
    end ;
end
end

```

ÖZGEÇMİŞ

1993’de Giresun’da doğdu. İlk ve Ortaöğretimini Kenan Evren İlköğretim okulunda tamamladı. Lise öğrenimini Giresun Anadolu Teknik Lisesinde tamamladı. 2012’de Giresun Üniversitesi Fen-Edebiyat Fakültesi İstatistik Bölümünü kazandı. Giresun Üniversitesi İstatistik ve Kariyer Topluluğunda dört yıl topluluk başkanlığı görevini yaptı. 2016’da Giresun Üniversitesinden bölüm birincisi olarak mezun oldu. Aynı yıl Giresun Üniversitesi Fen Bilimleri Enstitüsü Matematik ABD Yüksek Lisans programına başladı. Aktif olarak Bizim Toptan Satış Mağazalarında Yönetici Yardımcısı olarak görev almaktadır.